Deliverable 39

# FINAL REPORT

for

**Contract DAA-L01-97-C-0093**
**SBIR Topic A95-062**

# *NETWORK SIMULATION OF TECHNICAL ARCHITECTURE*

**September 30, 1998**

**Submitted to:**

**Telecommunications Division**
**Information Science and Technology Directorate**
**Army Research Laboratory**





**Submitted By:**

**PREDICTION SYSTEMS, INC.**
309 Morris Avenue
Spring Lake, NJ 07762
(732)449-6800
(732)449-0897
psi@predictsys.com

# TABLE OF CONTENTS

# 1. BACKGROUND

Fundamental to the Army's goal of establishing force structures for the twenty-first century is an imperative need to maintain interoperability across multiple telecommunications and information systems. The aggregate of these systems is the Army Battle Command System (ABCS). In order to set up a network that allows interoperable communications among users on the same or different command levels, a viable architecture is necessary. This architecture must be used to design, develop, and test systems in the context of the architecture. Simulation of the architecture and systems operating in the context of the architecture is a cost effective way of evaluating performance characteristics of the architecture.

The Army needs a tool to assist in designing technical architectures to guide the definition, design, and development of the Army battle command information transport that supports the goals of interoperability and flexibility to build a battle information infrastructure across all battle command systems. The information transport that supports the goals of interoperability will support seamless communications for all users on the battlefield, within and among the tactical, strategic, and sustaining base environments. To meet these needs, the Army Research Laboratories (ARL) initiated the Network Simulation of Technical Architecture (NSTA) project under the Small Business Inovative Research (SBIR) program.

In the early 1980's, PSI started developing a CAD approach to discrete event modeling and simulation that has cut large scale simulation life cycle costs by as much as an order of magnitude. This technology, based upon significant departures in technical concept and approach, provides ease of control and reuse of complex models. It has broken the barriers to modeling complex systems, yielding the very high resolution models required to insure validity. This is done by achieving model independence through graphically enforced architectual design rules and a fully automated user-friendly environment. This success has resulted in huge simulations that meet customer validity constraints and has prompted the desire for even higher levels of simulation integration.

As an extension to the above CAD approach to model design, PSI developed a Run-Time Graphics (RTG) system to depict the activities in simulations as they unfold and allow analysts to interact with simulations while they are running. The combination of these existing technologies has been used by the Army over the last decade to analyze, design, test, and evaluate its existing communication systems, e.g., MSE, SINCGARS, EPLRS, and satellite systems.

This existing technology supported most of the Army's needs in simulation, providing a sound basis upon which to build. But, certain improvements were needed to insure that this technology would meet future requirements for analyzing extremely large scenarios and extremely complex communication protocols and systems. To this end, ARL funded PSI under the NSTA project to make significant upgrades to its Run-Time Graphics (RTG) system. These upgrades allow users to create and modify huge model hierarchies, graphically - while simulations are running - in real-time. This new technology facilitates the ease with which users can interactively manipulate and control complex hierarchies of organizations and equipment using iconic models, and watch responses to system dynamics under various stress conditions using a braod selection of graphical depictions and visual instrumentation.

## 2. INTRODUCTION

The objective of the NSTA project is the development of a network modeling capability to simulate a battle command technical architecture to guide the definition, design, and development of the Army battle command systems. The PSI efforts thus far have been conducted in two phases. Phase I of the SBIR program provided for research into the graphical technologies necessary to insure successful completion of a Phase II effort. During Phase I, PSI analyzed, developed, and demonstrated the key technologies needed to support the new graphical facilities proposed.

As part of the Phase II effort, PSI produced detailed designs of an advanced version of its Run-Time Graphics system (RTG 7.0) based upon a hierarchical icon architecture. It then developed, and demonstrated the ability of this new system to support the deployment of aggregate operational facilities, including shelters, and the push-down of these facilities into basic entities. These entities include various suborganizations of people and equipment of different types, allowing users appropriate graphic interaction at each level in the hierarchy. In addition, the new system supports hierarchical icon movement where icons can contain sub-icons that move independently of each other but relative to the next level in the hierarchy, e.g., turrets on tanks turning while the tank is moving.

The tool that has been developed will help integrate various technologies including tactical multinet gateways, commercial standards and technologies (e.g., ATM/SONET, ISDN), high capacity local area networks, personal communications systems, small satellite platforms, ground terminals, direct broadcast satellite technology, interactive multimedia, and video teleconferencing. The tool can be used with defined sets of performance requirements and constraints to simulate an infrastructure that is flexible (facilitates force structure planning and dynamic reconfiguration), interoperable, and cost effective by taking advantage of commercial information technologies through adherence and use of open standards, protocols and products, and state-of-the-art telecommunications.

The hierarchical simulation run-time graphics system completes a large-scale discrete event simulation facility, for which a substantial, government-owned communications model base already exists, having evolved over the last 15 years. Commercial application of this development extends to areas such as real-time control of manufacturing processes, communication network equipment design, manufacturing plant design, process control system design, local, national, and international econometric modeling, business market modeling, world-wide commodity and currency stock and flow predictions, etc.

Under the NSTA project, PSI is providing training to qualified government personnel on the use of its General Simulation System (GSS) and Run-Time Graphics (RTG) system. It is also providing new licenses for existing clients and for those that attend training sessions, as well as licenses to the Army.

# 3. REVIEW OF FUNCTIONAL OBJECTIVES

Broad functional objectives were established for the Phase II effort that were based upon the success of the Phase I research. These are capabilities that are required by modelers and simulation developers supporting design and deployment of communication networks. This market audience covers communication equipment developers, users in the field, network planners, and trainers. Key functional objectives were the following:

- **Visual Interactive Modeling** - Create new models, visually, using interactive graphics, by interconnecting icons representing previously developed lower level models to form more complex model hierarchies. Change model parameters via menus and tables for prompted input values. Do this *while the simulation is running.*

- **Virtual Model Hierarchies** - Create hierarchies of models by attaching models into subhierarchies and subhierarchies into more complex hierarchies. Be able to change the hierarchical structures to test the results of each. Do all of this *while the simulation is running* as well as in an off-line Icon Library Management (ILM) drawing board facility.

- **Closed-Loop Experimentation** - Interact with real networks, making changes and watching the results in real-time, *while the simulation is running,* using convenient visual representations of instrumentation to depict the responses, dynamically.

- **Scenario Development and Analysis** - Interface with popular database management systems, e.g., DBase, FOXPRO, ACCESS, ORACLE, etc. for creating and maintaining large scenario databases. Interface with popular spreadsheet and statistical analysis packages, e.g., EXCEL, LOTUS, SAS, SPSS, etc., for performing data management and statistical analysis tasks.

The above objectives are aimed at supporting a new trend in modeling and simulation of complex technical architectures of networks. This trend is aimed at breaking up the overall modeling problem into layers that can be addressed by using different levels of skill. The growing need for rapid analysis of commmunication networks to support planning and evaluation far exceeds the skills necessary to support these efforts, and does not permit time for detailed modeling to insure validity when the need arises. To combat this problem, modelers have evolved shelves of ready-to-use submodels that can be reused easily in different simulation experiments. This is particularly true when performing Distributed Interactive Simulation (DIS) experiments, and when using simulation to support test planning and test augmentation.

## VISUAL INTERACTIVE MODELING

Figure 3-1 illustrates the recent evolution of skills to address rapid prototyping and analysis functions as supported by modeling and simulation. The major difference between the skill sets in this figure and those of the not-too-distant past is the two layers of modelers, namely the detail modelers and the higher level modelers. The separation between modelers and analysts has existed since the development of complex models became a significant software problem.

Now, because of the careful scrutiny of model validity, and the corresponding growth in model complexity, detail modelers must be familiar with the inner workings of a particular piece of equipment to insure sufficient accuracy of representation using sufficiently high model detail.

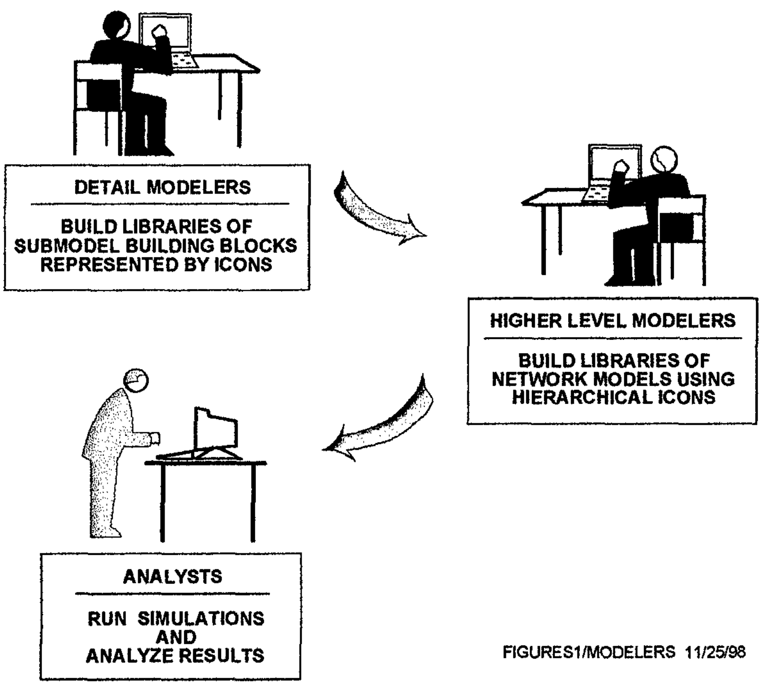


Figure 3-1. Hierarchical modeling to support organizational skills.

Higher level modelers are typically under great pressure to produce simulations under short deadlines to answer important questions of the moment. Typically, their only solution is to find existing models that are sufficiently accurate for the performance measures required by the immediate problem. Often there is time to perform modifications, but not cold starts. Thus, a model that has sufficiently high detail, and therefore a sufficiently wide range of validity, becomes a very valuable tool for reuse in many future simulation experiments to support rapid response analysis.

## VIRTUAL MODEL HIERARCHIES

When modeling communication networks, if models of equipment, e.g., switches, digital radios, etc., are designed along physical lines with independent components, then these models are candidates for ease of modification at different protocol layers. Over the past twelve years, PSI has been involved in heavy reuse of models developed for totally different projects. This has led to careful development of detailed models to insure their ease of reuse in new simulation experiments.

Out of this experience has evolved a new technology - one of creating hierarchical symbolic models that can be interconnected to form higher level models. The actual interconnection is analogous to connecting pieces of equipment together to form a network. Using this technology, higher level modelers can create network models using icons that get connected graphically. If the models behind the icons are built correctly, then these interconnections are fed to those models - while the simulation is running, just as the equivalent interconnections are recognized by the equipment, and they proceed to operate accordingly. The Phase II Design Document contains the design that supports this capability.

## SCENARIO DEVELOPMENT AND ANALYSIS

PSI has implemented facilities to support the Standard File Interface (SFI) specification that was developed in conjunction with a number of government agencies and contractors. SFI permits users to interchange files easily, and read and write these files using standard library modules available at no cost from a number of sources. Over the past two years, a new version of the SFI specification has evolved that allows more flexible interfaces to popular database management systems, e.g., DBase, FOXPRO, ACCESS, ORACLE, etc., and spreadsheets, data and statistical analysis packages, e.g., EXCEL, LOTUS, SAS, SPSS, etc. PSI has developed automatic facilities for supporting this new standard, and has used this facility in a new version of GSS as part of this Phase II effort. This permits ease of scenario development, and data and statistical analysis.

Interactive instrumentation will still exist as part of the new Run-Time Graphics system. This means that modelers can provide handy instruments for analysts that can take measurements at nodes or between nodes while the simulation is running. This has proved to be a very powerful analysis tool, allowing the analyst to sample various measures of performance at any time during a simulation scenario, and to dynamically change what is being measured based upon the results at that time. The instrumentation facility coupled with the ability to blink, and change the style, thickness, and color of network links and nodes, as traffic or other properties change during the course of a simultion, provides for excellent visual representations of what is happening in complex network simulations. This has afforded modelers the ability to show the dynamics of complex protocols, e.g., MSE flood search, under different stress scenarios so that people with less technical backgrounds obtain a good feeling for the trade-offs of different technologies.

# 4. TECHNICAL OBJECTIVES

The technical objectives for this effort are listed below. They were derived based upon use of GSS and RTG by various government agencies over the past 12 years to simulate very large scale communications systems. The existing multi-million dollar shelf of high quality, well documented GSS models, owned by the government, in continuous active use - running simulations to perform analyses and answer questions of current interest - provided an excellent resivoir of knowledge upon which to base the RTG improvements.

For example, most modelers think of graphics as an output facility, to show what has happened in the simulation - after the fact. Some use graphics to aid in model development - requiring compilation every time a model is changed. However, the use of RTG - *while the simulation is running* - is now being appreciated by a growing number of PSI's clients. This has led to the objectives summarized below for raising RTG to a complete hierarchical modeling run-time graphics facility that provides the Army with a unique level of ABCS simulation support readiness for many years into the future. Specifically, it supports:

- Drawing icons using an RTG drawing board facility.
- Transforming icons from the drawing board into the RTG icon library with rotation and scaling.
- Transforming icons from the RTG icon library into a particular simulation as a normal or initial view, including rotation and scaling.
- Transforming icons from the RTG library into hierarchical iconic models for a particular simulation.
- Providing the modeler with generalized coordinate systems to be used in his specific simulations by his models.
- Performing transformations on hierarchical icons at various levels in the hierarchy, while viewing them at different levels in the hierarchy.
- Panning and zooming on scenes (changing the viewpoint) in a simulation as hierarchical icons are undergoing their own transformations.
- Providing easy to use implementations of the above requirements so that the modeler can think in terms of his own problem and coordinate systems, with minimal effort to perform transformations.
- Providing speed and accuracy of the operations in the implementation.

Motivation and examples for a hierarchical icon graphics facility are provided in the next two sections. Figures 4-1 and 4-2 on the next pages illustrate the basic hierarchical icon approach.

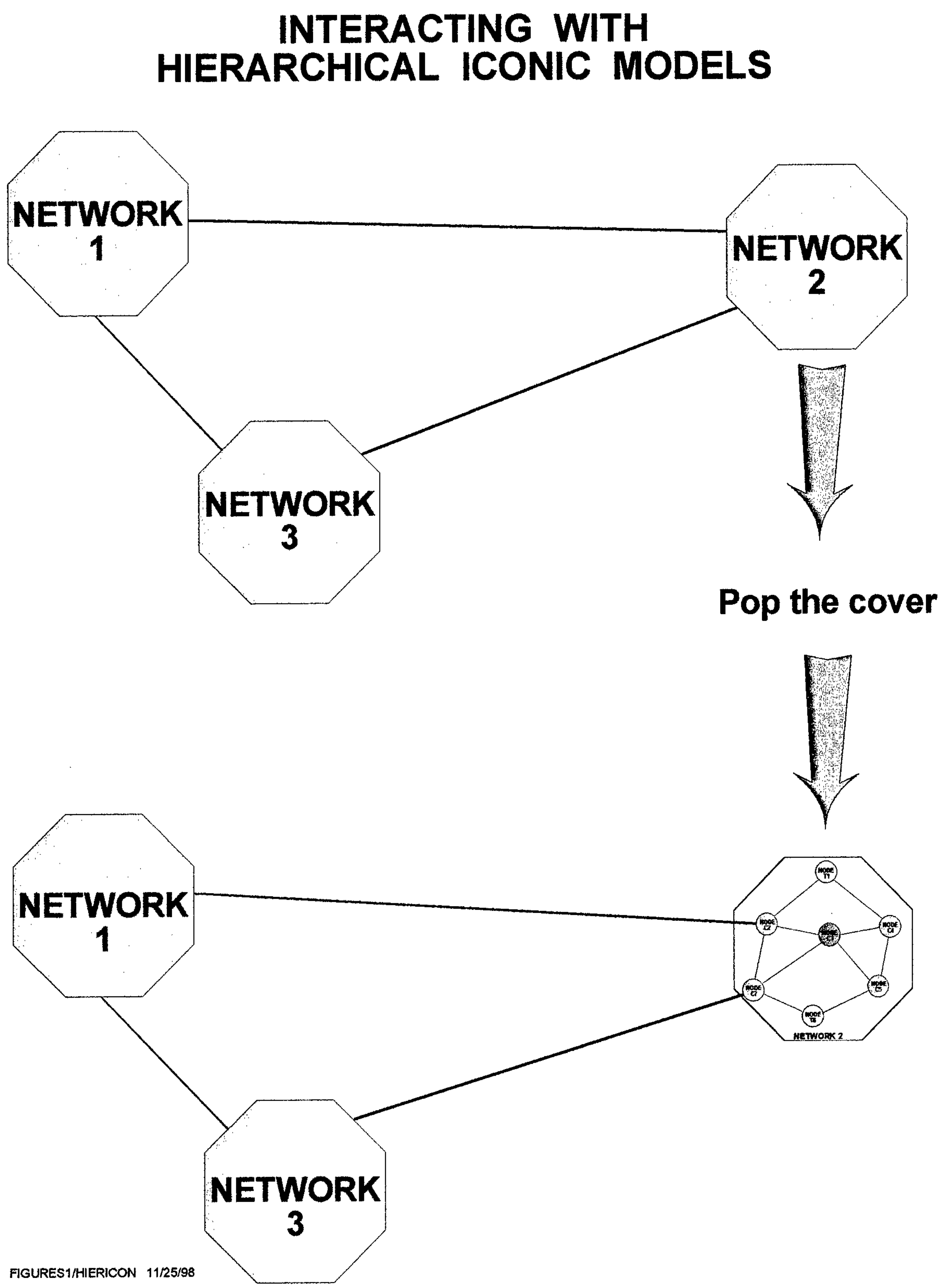


Figure 4-1. Popping the cover off of a hierarchical icon.

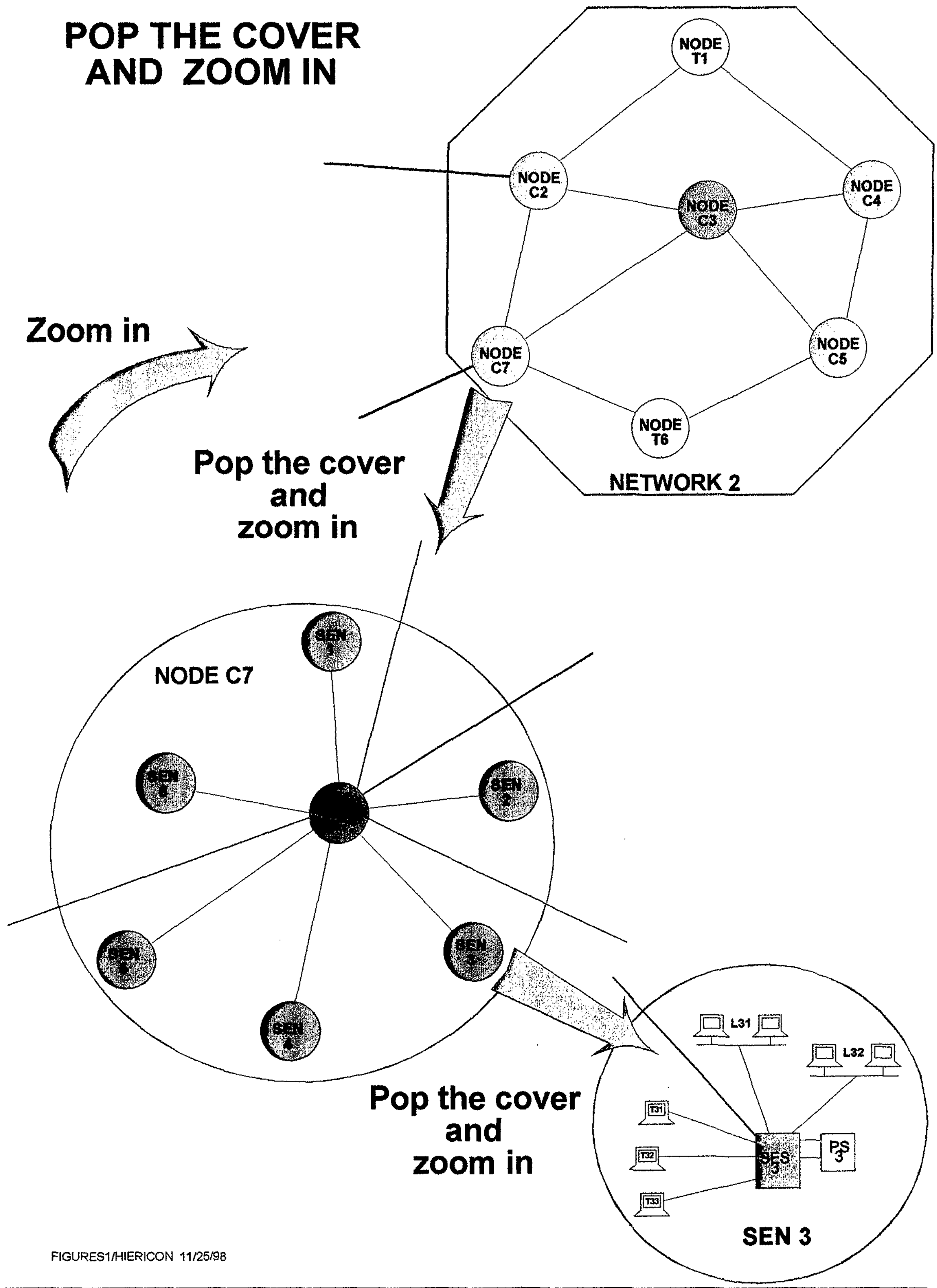


Figure 4-2. Popping the covers and zooming in on the details.

**The Requirement For Hierarchical Icons**

Why are hierarchies of icons important in simulations, particularly simulations of communication and control systems? Because the real entities being simulated are constructed of hierarchies of elements. To use an example, consider the communication network depicted in Figure 4-1. At the top of the figure, a simple three node network is shown consisting of three interconnected icons. These icons each represent a network themselves. To understand what is in NETWORK 2, we can pop the cover off the NETWORK 2 icon, and see the subnetwork at the next layer.

When we first pop the cover, it is difficult to see the subnetwork comprising NETWORK 2. So in Figure 4-2, we show the effects of zooming in to have a closer look at the subnetwork. It contains 7 nodes, 5 central nodes (e.g., C2) and 2 tandem nodes (e.g., T1). Nodes C2 and C7 are connected to the other higher level networks outside the subnet. Similarly, if we want to see what is in NODE C7, we can pop the cover off that node and zoom in to see the next layer down. It contains 6 small extension nodes (SENs) connected to a node central switch NCS 7.

Finally, if we pop the cover off of SEN 3, we see the subnetwork at that node. It contains a small extension switch (SES 3), a packet switch (PS 3), three terminals (T31, T32, and T33), and 2 local area networks (L31 and L32). Depending upon the simulation, this may be the bottom layer for iconic representation available to the user graphically. The icons at the lowest layer are elementary icons. Hierarchical icons are made up of elementary icons or lower level hierarchical icons.

Given that we can deploy icons representing the particular real elements or hierarchies of interconnected elements, then we can greatly simplify dealing with the complexity of large networks. For example, there are 7 elements in node SEN 3 represented by 7 icons. If this is the size of an average small extension node, and the other subnetworks are average sizes, then the three icon network at the top of Figure 4-1 represents 1000 elements. Since the three icon network can easily be represented by a single icon, we can deploy a network of 1000 elements just by deploying a single icon - *while the simulation is running.*

We now have the option of looking at any of the detailed layers at any node or subnode. We need not look at 1000 icons to see the picture or any details of interest. This can greatly simplify the analyst's job when working with realistic deployments in the field. In addition, it significantly reduces screen clutter and increases the speed of the draw.

In addition to viewing the picture, we also want to interact with the network, changing it as we see fit - while the simulation is running. We may want to move more terminals into SEN 3. We may want to take these terminals from SEN 2. If the models are designed properly, the simulation will proceed to represent what would happen if this were done in a real exercise. One could also disconnect and reconnect nodes at any layer in the hierarchy, while the simulation is running. When this happens, individual subnodes may have to be redeployed, to insure connectivity, just as they would in a real exercise.

## Creating And Interacting With Simulations Using Iconic Models

Just as we can pop covers and zoom into more detailed iconic representations of physical systems, we may also want to interconnect icons to build networks of new systems - while the simulation is running. The only requirements are that the icons be available to the simulation, and the model that gets invoked when a particular icon is deployed represents what the user expects.

PSI has had this facility for more than a decade with the prior RTG system for "flat" network structures. However, when an analyst has created a new network, say at the SEN level in Figure 4-2, he would like to cover up the details using a SEN 3 icon. Having created the desired SENs, one can then easily create a NODE C7 type network using the NCS and SEN icons. Similarly, one can create NETWORK 2 type subnetworks using the C and T NODE icons.

With the new RTG system, higher level models can be created just by interconnecting icons - while the simulation is running. These models can also be stored away for future use, simply as a scenario. This is critical for saving time when dealing with complex networks, particularly when one must look at some results prior to accepting the physical network architecture. A good example is a network using radio transmission to connect nodes. In this situation, the analyst wants to obtain some form of connectivity before accepting the network as representative of the real world situation. This can be adjusted graphically until it is ready for use. Then the scenario can be stored for future use as an initial deployment at another time.

# 5. TECHNICAL EFFORTS

The major research and development efforts performed under the Phase II contract were those for developing the hierarchical Run-Time Graphics (RTG ) system and the ICON Library Managment (ILM) system. Depending upon the context, references to RTG may imply inclusion of the ILM system. This is particularly true with regard to distribution of the RTG system and installation package which includes the ILM system as part of the package. Because these are both significant pieces of software, they are described in separate sections of both the Phase II Design document and the RTG Users Manual. This Final Report merely summarizes the effort and those documents, and the reader is referred to the Users Manual for functional details, and the Design Document for implementation details. We will start by describing the ILM system.

## ICON LIBRARY MANAGEMENT (ILM) SYSTEM AND DRAWING BOARD

Figure 5-1 is a top-level block diagram of the ILM system. The user creates new icons and modifies existing icons in the icon library. It provides the facilities to create and edit icons using primitive symbols, e.g., lines, triangles, rectangles, circles, ellipses, polygons, etc., or previously built icons. The user selects line styles, thickness, and colors for the individual primitive symbols, allowing multi-colored filled icons to be drawn. In the case that the user desires to change the color of an icon, he can specifiy that certain elements of an icon have a variable color, and the color property can be changed while the simulation is running. Icons are stored in the icon library for future use in the drawing board or in a simulation.

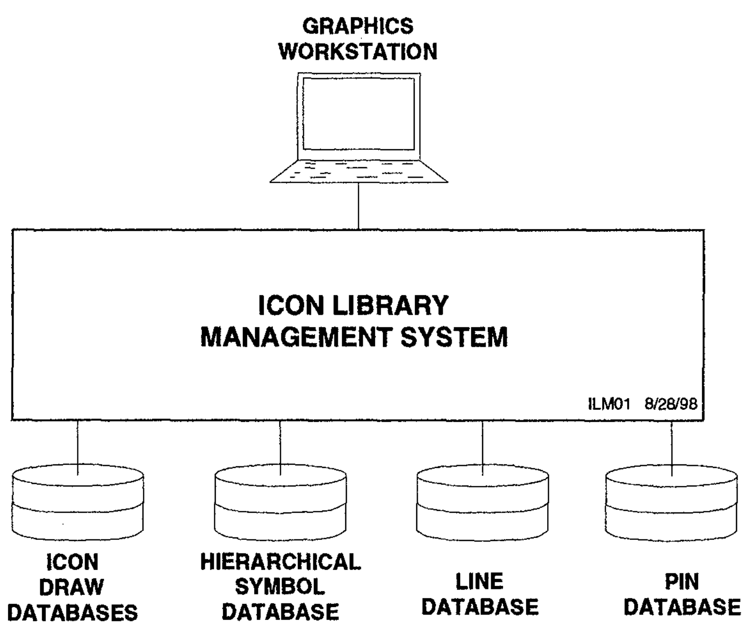


Figure 5-1. ICON Library Management (ILM) system.

Hierarchical icons can be created by *attaching* two or more icons to a higher level icon. These icons can also be connected by lines using pin connections, or simply by connecting to the center of the icon.

Figure 5-2 illustrates the use of the drawing board to create a hierarchical icon. The group of seven icons at the bottom of the screen will be represented by the icon C1 at the top. When the cover on C1 is lifted, the group can be displayed with or without the outline of C1 as shown at the bottom, and with or without the pins showing. The connector lines become part of the hierarchical icon.

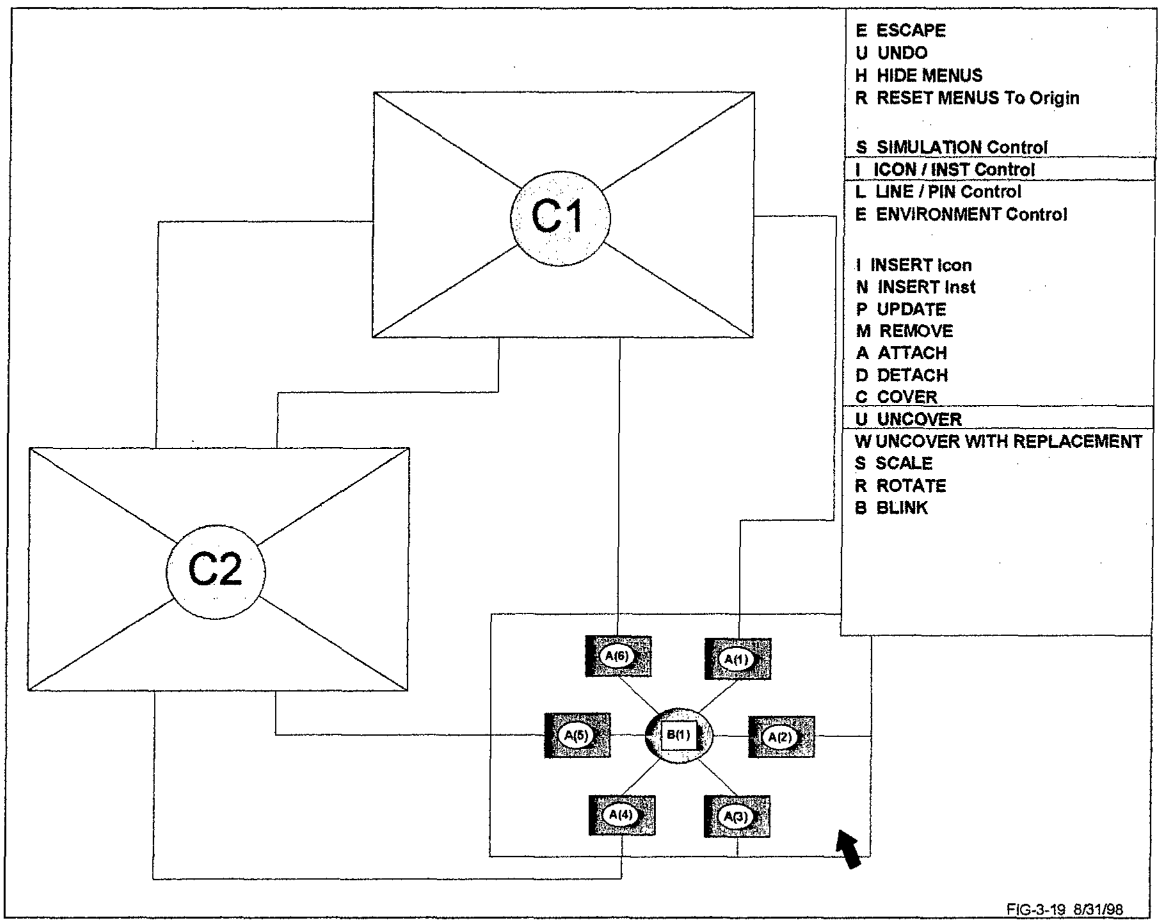


Figure 5-2. Creating hierarchical icons using the drawing board.

This facility allows users to build up hierarchies of iconic representations of networks, where models created in GSS can be put behind the icons.

## RUN-TIME GRAPHICS (RTG) SYSTEM

Figure 5-3 is a top level block diagram of the RTG system, and its interfaces to the General Simulation System (GSS). These two systems run as separate tasks in a multi-tasking operating system, e.g., they are separate UNIX processes. This provides maximum efficiency for handling graphical inputs via the mouse. The tasks communicate via the two intertask resources shown between the tasks. The user can interface directly with the simulation via the same workstation that is used for the Run-Time Graphics.

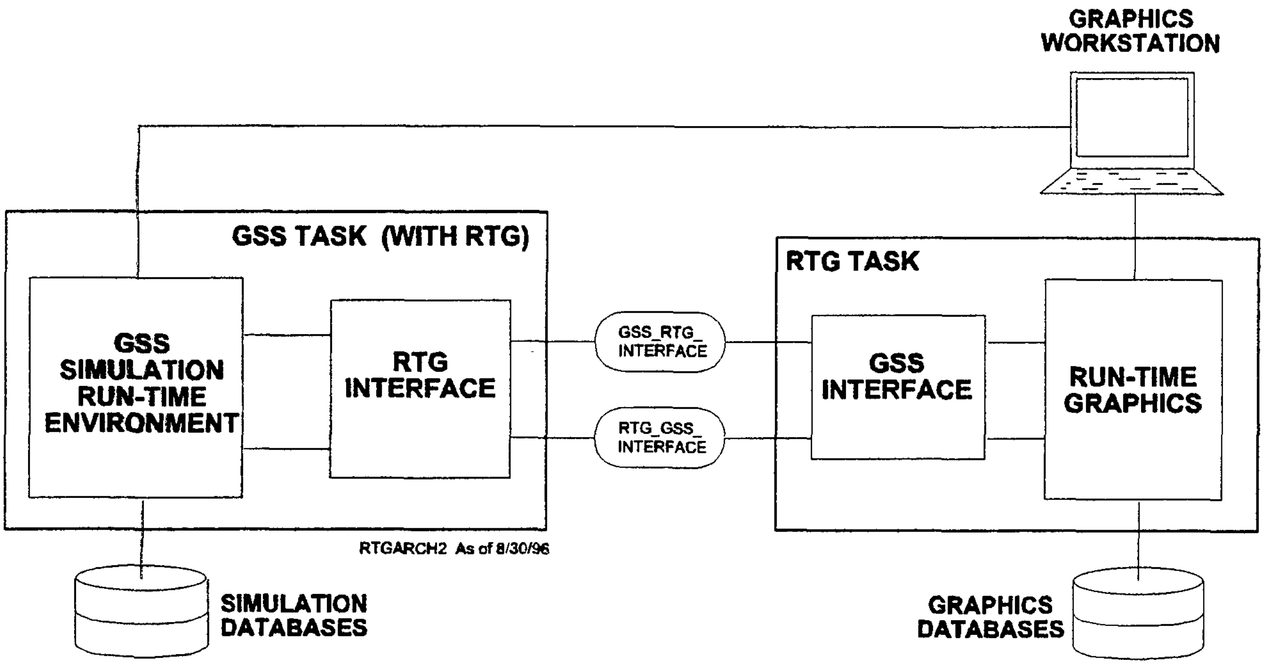


Figure 5-3. Run-Time Graphics (RTG) system shown with GSS.

When using RTG, the user will generally interface with the simulation using the graphics facilities. However, there are occasions, especially when interfacing with real-time devices or communication channels that the user will want to interface directly with the simulation. As indicated above, the reader is referred to the GSS and RTG User's manuals for more details.

As a brief synopsis, users can create graphical representations of simulation dynamics by issuing commands directly from GSS, e.g., to insert icons, move them around, change their sizes, colors, and other attributes, interconnect them with lines, draw background overlays, e.g., terrain contours, foliage, roads, etc. to depict detailed Army type maps using digitized NIMA databases. These same databases can be used for propagation calculations to determine radio signal-to-noise ratios at receivers, with jammer noise powers computed from dynamic jammer waveforms eminating from jammers in moving air or ground units.

In addition to issuing commands from the GSS side (i.e., the simulation) via statements imbedded in the models, the user can also interact directly with the graphics at the workstation using the mouse and the keyboard. Icons and instruments can be inserted, moved, and removed. Their sizes, colors, and other properties can be changed. They can be interconnected by lines, etc. When this is done via the graphics workstation, the pertinent information is sent to the simulation, and the models can be used to effect the proper simulation responses.

Prior to the new RTG, existing graphics systems provided a "flat" view of any scenario, i.e., either the icons representing the models are visible on the display or they are not. This leads to screen clutter and slow draws when large quantities are active in a scenario. This also makes it difficult to control the natural grouping of units as normally occur in the Army's organization.

The results of this Phase II effort provide a hierarchical icon capability, so that many layers of hierarchy can be selected for visibility at different points in the hierarchy. Figure 5-4 illustrates a commercial U. S. phone network at a point in the simulation where the modeler desired to uncover and zoom in on a particular data center.

For this data center, elementary icons depicting communications multiplexors, disk drives, a mainframe CPU, and instrumentation are now in view. The user can drill down to any lower layer at any desired points in the hierarchy to make the details visible. Or the user can cover up unwanted details by performing the reverse functions. Experienced RTG users have ranked the need for this capability as very high for a number of years. This is because GSS users have been able to build large scale simulations with relative ease compared to other simulation environments. In doing so, they have all encountered the difficulties associated with building large but realistic scenarios, as well as trying to *see* what is going on while a large scale simulation is running, to assess validity.

This hierarchical icon capability removes the major impediments to solving these two problems. In addition, it provides the facilities needed to allow the higher level modelers to build networks using hierarchical icon models, without having to build the details. This will be a major breakthrough in carrying out rapid prototyping and analysis tasks with small budgets and short response times.

## TEST SIMULATIONS AND USER FEEDBACK

PSI wants to ensure that what has been developed meets the real needs of modelers and analysts who are assessing the technical architectures of Army communications networks, and to insure they have the best tools to support their efforts for the forseeable future. To this end, we allocated resources to transition existing popular simulations of Army communication systems and DIS experiments to demonstrate to existing users that this effort has met all of its objectives.

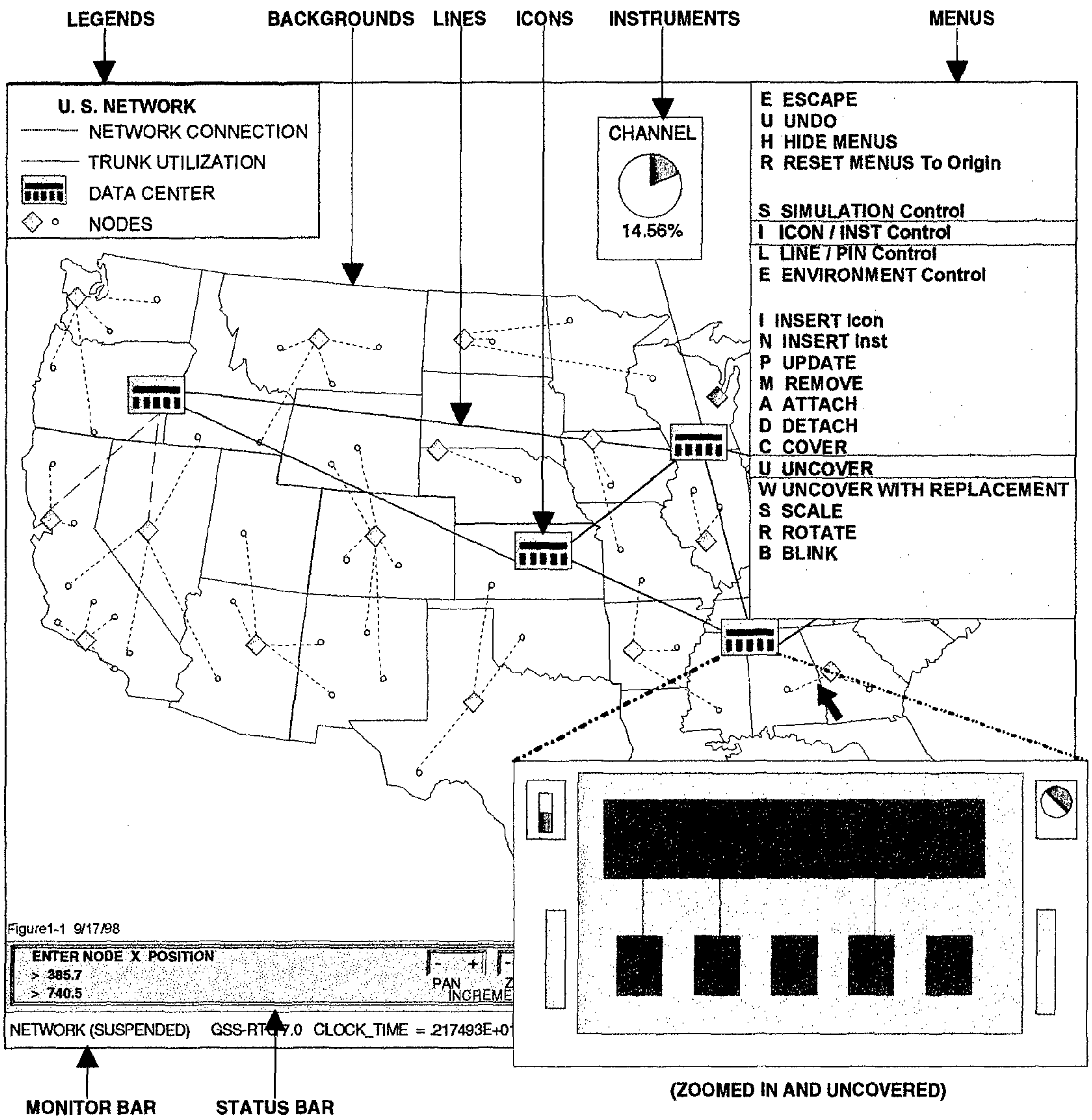


Figure 5-4. Hierarchical RTG simulation of a U. S. phone network.

Two complex GSS simulations owned by DoD at Ft. Monmouth were selected as part our test bed for human-interfaces as well as to augment regression testing when changes are made. These simulations are representative of many GSS simulations owned by other clients. Most important, they provided excellent test beds for the hierarchical iconic models, having a similar structure as that illustrated in Figures 4-1 & 2. These two simulations contain ample quantities of hierarchical levels and different hierarchical structures to test the user interfaces from a human engineering standpoint as well as test the systems from a total quality and reliability standpoint. They also provided valuable examples of how one can create the detailed models so that the higher level modelers can create rapid prototype simulations and support quick analyses. At the same time, PSI issued initial releases to consultants as friendly users for testing the human interfaces and providing valuable feedback regarding the quality of the initial product.

## UPGRADES AND REGRESSION TESTING

Based upon many prior experiences with software product releases, particularly when complex graphical user interfaces are involved, sufficient time had to be alloted for testing, upgrading, and regression testing to ensure a quality product. To accomplish this, as feedback was received from PSI's "user" consultants, as well as its in-house staff, PSI upgraded the initial release and performed regression testing using the two simulations to augment the large body of test cases that have been accumulated over the years. This was done to prepare for the first general release of the new systems.

## INSTALLATIONS AND TRAINING

Upon completion of the first general release, PSI is offering training courses for existing clients and personnel at key Army sites so that critical users, e.g., ARL, CECOM, AMSAA, and DISA can be brought up to date. This will provide credibility for the new approach. Training is planned for a central location, near CECOM, Ft. Monmouth.

## BREADTH OF APPLICATIONS

Although the examples used in this report are communication system oriented, there is no limitation on applications of GSS and RTG to other areas of dynamic simulation. In fact, models have been created for intelligence systems, including sensor tasking, objects and their motion, various sensors and platforms in motion; air defense systems, including high and low speed aircraft, radar, targets, and attrition; fire support systems, including observers, target detection and identification, command and control, weapon systems, kill measures, etc.

Because of the ability to interact with the simulation while it is running, and the ability to tie the simulation clock to the real-time clock (or some fraction or multiple thereof), these facilities can be used for live hardware-in-the-loop testing and specifically multiple simulations interacting as part of a Distributed Interactive Simulation (DIS) or Higher Level Architecture (HLA) network experiment.

Neither the concept nor the capabilities provided are limited to things that connect externally, e.g., switches, terminals, etc. There is no reason that protocol layers and other more elementary components, e.g., data coding and compression modules, can't be combined using icons. One must simply design models that support the type of interactive interconnection described above. A modeler that designs models in this way can provide a library of iconically represented models to analysts and engineers that can be interconnected to build a higher level model graphically - while the simulation is running. Additional facilities are described in the RTG Users Manual, e.g., pin connections, to assist in this process.

# 6. RESULTS OF THE PHASE II EFFORT

This SBIR Phase II effort was aimed at implementation of a hierarchical iconic modeling facility that can be used to structure and restructure both models and scenarios, interactively, *while simulations are running*. When incorporated into the GSS Run-Time Graphics system, this tool provides an open environment in which modelers can develop their own tailored models and share them easily with anyone. These models can be copied and modified by different organizations for use in different applications while being protected from unwanted change at the same time.

During this effort, PSI built and demonstrated advanced run-time graphics facilities for hierarchical icons which support the deployment of an aggregate facility, such as a shelter, and the push-down of this facility into basic entities, such as various terminals, computers, multiplexors, switches and radios of different types, with appropriate graphic interaction at each level in the hierarchy. In addition, hierarchical icon movement, where icons can contain sub-icons that move independently of each other, but relative to the next level in the hierarchy, was demonstrated.

These facilities allow the user to redefine the hierarchies interactively - while the simulation is running. This is demonstrated by the detachment and attachment of subhierarchies at any point in the original hierarchy. In addition, the ability to save the updated hierarchies for future use in different simulations was also demonstrated. This feature greatly simplifies the development and export/import of different scenarios.

The full functionality of this system is documented in the new Run-Time Graphics (RTG) system Users Manual, including the Icon Library Management (ILM) system. System design details are provided in the RTG and ILM Design Documents. In addition, a System Tutorial has been provided to support training for new users. Therefore, only the highlights have been presented here for convenience. PSI has completed the design and implementation of the hierarchical icon facility as it will fit into an overall simulation run-time graphics system. We have implemented the required hierarchy of RTG states, the use of text boxes, the use of menus, and the overall control window facilities required to support a complete graphical simulation facility. We have completed the database to support moves of hierarchical icons from one spot in the hierarchy to another.

We have also built, tested, and subsequently modified and retested all of the mouse pick and select modules. This has been accomplished through extensive testing of drawing responses to fast mouse motion when creating select boxes or dragging icons. We have implemented the overall user interface in terms of states and transitions from state to state. We have implemented a scaleable font alphabet to support zooming, rotating and stretching of icons whose text labels remain in proportion and position with the icons. We have completed the handling input events from the simulation as well as from the user at the interactive workstation.

Very little of the above efforts could have been performed in the period of performance of this SBIR Phase II contract had PSI not had substantial experience in the development of graphical interfaces to support communication network simulations in a complex and sophisticated customer environment. Having built and supported upgrades to the prior RTG system over the past nine years, PSI has gained a wealth of knowledge about what the users want, how to achieve platform independence in a sophisticated graphics environment, and how to design the software architectures so that they are flexible to accommodate change as new functions and features are desired.

By taking a totally new technological approach, GSS users have had the ability to construct and modify their models *while the simulation is running.* This allows one to observe the changing results, and make informed decisions regarding how to restructure a model based upon the unfolding scenario. This dynamic operating environment eliminates the need to "recompile" a model before running another simulation, or to have to wait to review results graphically after the simulation has ended.

## 7. SUMMARY & CONCLUSIONS

In this project, we have demonstrated that, by incorporating the hierarchical icon capability into the GSS Run-Time Graphics system, it is extremely easy to construct and vary complex network architectures and their environments and get simulation results graphically. This helps analysts build and control the very large scenarios needed to construct a total communications picture of the battlefield. Hardware horsepower has grown to support these experiments. This effort has put the GSS-RTG tools used by the Army well ahead of the hardware. This is precisely what experienced Army modelers now need and want.

By changing the network interactively, i.e., dynamically while the simulation is in progress, one obtains immediate answers and feedback regarding network designs. This allows analysts to construct experiments on the fly. It provides a modeling and simulation technology that directly supports the requirements for Distributed Interactive Simulation (DIS) experiments right now – with or without Higher Level Architecture (HLA) facilities. This interactive graphical DIS facility has been demonstrated at various conferences for command and control in Tactical Operations Centers (TOCs) as well as for communication equipment architecture and deployment. The facility built under this SBIR Phase II program has put the Army significantly ahead of any other organization in modeling and simulation of communication and control systems.

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*



| 1. AGENCY USE ONLY (*Leave Blank*) | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | | |

4. TITLE AND SUBTITLE

NETWORK SIMULATION OF TECHNICAL ARCHITECTURE

5. FUNDING NUMBERS

DAAL01-97-C-0093

6. AUTHOR(S)

William C. Cave and Robert E. Wassmer

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Prediction Systems, Inc.
309 Morris Avenue Suite G
Spring Lake, New Jersey 07762

8. PERFORMING ORGANIZATION REPORT NUMBER

PSI 98002

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

ARMY RESEARCH LABORATORY
ALC PROCUREMENT DIVISION
2800 Powder Mill Road
Adelphia, MD 20783-1197

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

NONE

12a. DISTRIBUTION/AVAILABILITY STATEMENT

UNLIMITED

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

This SBIR Phase II effort was aimed at implementation of an interactive graphics facility for network modeling to simulate battle command technical architectures to guide the definition, design, and development of the Army Battle Command System (ABCS). PSI delivered a hierarchical iconic modeling facility that can be used to structure and restructure both models and scenarios, interactively, *while simulations are running*. This tool provides an open environment in which modelers can develop their own tailored models and share them easily with anyone. These models can be copied and modified by different organizations for use in different applications while being protected from unwanted change at the same time.

During this effort, PSI built and demonstrated advanced run-time graphics facilities for hierarchical icons which support the deployment of an aggregate facility, such as a shelter, and the push-down of this facility into basic entities, such as various terminals, computers, multiplexors, switches and radios of different types, with appropriate graphic interaction at each level in the hierarchy. In addition, hierarchical icon movement, where icons can contain subicons that move independently of each other, but relative to the next level in the hierarchy, was demonstrated.

14. SUBJECT TERMS

DISCRETE EVENT SIMULATION    HIERARCHICAL RUN-TIME GRAPHICS
MODEL DEVELOPMENT GRAPHICS    COMMUNICATIONS MODELS
ICON HIERARCHIES

15. NUMBER OF PAGES

18

16. PRICE CODE

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNLIMITED | UNLIMITED | UNLIMITED | UNLIMITED |